"A data quality system"

INTRODUCTION

Field of the Invention

The invention relates to a data quality system.

Prior Art Discussion

Data quality is important for companies maintaining large volumes of information in the form of structured data. It is becoming an increasingly critical issue for companies with very large numbers of customers (for example banks, utilities, and airlines). Many such companies have already, or are about to, implement customer relationship management (CRM) systems to improve their business development. Effective operation of CRM systems involves drawing data from a range of operational systems and aggregating it on a customer-by-customer basis. This involves a large degree of data matching based on criteria such as customer identification details. Such matching and associated operations are often ineffective because of bad quality data. The data quality problems which often arise include:-

empty fields;

lack of conformity, such as the letter "H" in a phone number field;

lack of consistency across fields of a record such as "customer status = live" and "last invoice date = 20/01/99";

lack of integrity of field values; and

duplicates.

In more detail, data matching difficulties arise from (a) the multitude of different ways in which two equivalent sets of data can differ, and (b) the very large volumes of data generally involved. This means that carrying out the task manually is impossible or

hugely costly and defining a finite set of basic matching rules to automate the process is extremely difficult. As organisations collect more data from more sources and attempt to use this data efficiently and effectively they are encountering this problem more frequently and the negative impact is growing.

It is therefore an objective of the invention to provide a data quality system to improve data quality.

SUMMARY OF THE INVENTION

According to the invention, there is provided a data quality system for matching input data across data records, the system comprising:-

means for pre-processing the input data to remove noise or reformat the data,

means for matching record pairs based on measuring similarity of selected field pairs within the record, and for generating a similarity indicator for each record pair.

In one embodiment, the matching means comprises means for extracting a similarity vector for each record pair by generating a similarity score for each of a plurality of pairs of fields in the records, the set of scores for a record pair being a vector.

In another embodiment, the vector extraction means comprises means for executing string matching routines on pre-selected field pairs of the records.

In a further embodiment, a matching routine comprises means for determining an edit distance indicating the number of edits required to change from one value to the other value.

In one embodiment, a matching routine comprises means for comparing numerical values by applying numerical weights to digit positions.

In another embodiment, the vector extraction means comprises means for generating a vector value between 0 and 1 for each field pair in a record pair.

In a further embodiment, the matching means comprises record scoring means for converting the vector into a single similarity score representing overall similarity of the fields in each record pair.

In one embodiment, the record scoring means comprises means for executing rule-based routines using weights applied to fields according to the extent to which each field is indicative of record matching.

In another embodiment, the record scoring means comprises means for computing scores using an artificial intelligence technique to deduce from examples given by the user an optimum routine for computing the score from the vector.

In a further embodiment, the artificial intelligence technique used is cased based reasoning (CBR).

In one embodiment, the artificial intelligence technique used comprises neural network processing.

In another embodiment, the pre-processing means comprises a standardisation module comprising means for transforming each data field into one or more target data fields each of which is a variation of the original.

In a further embodiment, the standardisation module comprises means for splitting a data field into multiple field elements, converting the field elements to a different format,

removing noise characters, and replacing elements with equivalent elements selected from an equivalence table.

In one embodiment, the pre-processing means comprises a grouping module comprising means for grouping records according to features to ensure that all actual matches of a record are within a group, and wherein the matching means comprises means for comparing records within groups only.

In a further embodiment, the grouping module comprises means for applying labels to a record in which a label is determined for a plurality of fields in a record and records are grouped according to similarity of the labels.

In one embodiment, a label is a key letter for a field.

In another embodiment, the system further comprises a configuration manager comprising means for applying configurable settings for the pre-processing means and for the matching means.

In a further embodiment, the system further comprises a tuning manager comprising means for refining, according to user inputs, operation of the record scoring means.

In one embodiment, the tuning manager comprises means for using a rule-based approach for a first training run and an artificial intelligence approach for subsequent training runs.

DETAILED DESCRIPTION OF THE INVENTION

Brief Description of the Drawings

The invention will be more clearly understood from the following description of some embodiments thereof, given by way of example only with reference to Fig. 1, which is a block diagram illustrating a data quality system of the invention.

Description of the Embodiments

Referring to Fig. 1, a data quality system 1 comprises a user interface 2 linked with a configuration manager 3 and a tuning manager 4. A data input adapter 5 directs input data to a pipeline 6 which performs data matching in a high-speed and accurate manner. The pipeline 6 comprises:

a pre-processor 7 having a standardisation module 8 and a grouping module 9,

a matching system 11 comprising a similarity vector extraction module 12 and a record scoring module 13.

The output of the pipeline 6 is fed to an output datafile 15.

The system 1 operates to match equivalent but non-identical information. This matching enables records to be amended to improve data quality.

The system 1 ("engine") processes one or multiple datasets to create an output data file containing a list of all possible matching record pairs and a *similarity score*. Depending on the needs of the user the engine can then automatically mark certain record pairs above a specified score as definite matches and below a specified score as non-matches. Record pairs with scores between these two thresholds may be sent to a user interface for manual verification.

There are a number of discrete activities within the matching process. These can be grouped into two separate phases: – pre-processing and matching

Pre-processing

In the pre-processing phase all data records are read sequentially from the data input adapters 5. Firstly each record is fed to the standardisation module 8 where a range of different routines are applied to generate an output record which can be matched more effectively with other records. Each record is then fed to a grouping module 9. In this process labels are attached to each record to enable it to be easily and quickly grouped with other similar records. This makes the downstream matching process more efficient as it eliminates the need to compare records which are definitely non matches. Following the grouping process the output record (transformed and labelled) is written to the pre-processed datafile.

Matching

In the matching phase, each record is read in sequence from the pre-processed dataset 10. It is then compared to each similar record in the dataset – i.e. records within the same group. The comparison process involves:

1. Similarity Vector Extraction: This involves comparing individual fields within a record pair using matching algorithms to generate a similarity score for each pair of fields. Data element scoring is carried out on a number of field pairs within the record pair to generate a set of similarity scores called a similarity vector.
2. Data record Scoring: Once a similarity vector has been produced for a record pair by a series of data element scoring processes, the data record scoring process converts the vector into a single similarity score. This score represents the overall similarity of the two records.

The pair of output records is then written to the output datafile along with the similarity score. The matching phase then continues with the next pair of possible matching pairs.

To achieve high accuracy matching, the setup of the modules is highly specific to the structure and format of the dataset(s) being processed. A key advantage of the engine is built-in intelligence and flexibility which allow easy configuration of optimum setup for each of the modules. Initial setup of the four processing modules is managed by the configuration manager 3 and the tuning manager 4.

Standardisation ("Transformation") Module 8

The aim of the transformation process is to remove many of the common sources of matching difficulty while ensuring that good data is not destroyed in the process. This is done by transforming the individual elements of a record into a range of different formats which will aid the matching process. Each data field in a record is transformed into a number of new data fields each of which is a variation of the original.

Each data record is read in turn from the adaptor 5. Each field within a record is processed by applying a number of predefined transformation routines to the field. Each transformation routine produces a new output data field. Thus, an output record is produced containing a number of data fields for each field in the input record. Field transformation routines include:

- Splitting a data field into multiple fields, for example splitting street address into number, name and identifier.

- Converting field elements to other format using conversion routines, for example:
  - Converting to uppercase.
  - Converting to phonetic code (Soundex).
  - Convert to abbreviated version.
  - Convert to standardised format (e.g. international telephone codes).
  - Convert to business-specific version.

- Removal of characters from within data field, for example:

- Removal of spaces between specified elements.
- Removal of specified symbols from between specified elements (e.g. punctuation marks / hyphens).

- Replacement of element with an equivalent element selected from an equivalence table, for example:
  - Replacement of nickname / shortened name with rootname.
  - Replacement of Irish/foreign language place or person name with English equivalent.
  - Replacement of standard abbreviations with root term (st. to street, rd. to road etc.).
  - Replacement of company name with standardised version of name.

The transformation module 8 is capable of carrying out a user-defined number of transforms such as those above to each input data field and generating a user-defined number of output fields for each input field. The transforms required for each field type may be configured by:

- Selecting from a menu of default transformation configurations (set of routines) predefined for use with a particular field type of a particular structure/format/quality level.
- Developing new configurations for each data field / element from a menu of transformations such as those above.
- Developing new configurations for each data field / element using bespoke transformations input by the user – probably combined with some predefined transformations.

In batch matching projects the transformation process is carried out on the whole database before any matching is done. A new data file of transformed elements is then created for use in the matching process. This saves time by ensuring that the minimum

number of transformations N are carried out (where N = number of records in the database) rather than the potential maximum number of transformations NxN. However in realtime search and match operation the transformation process is carried out directly before the matching process for each record.

The following is a transformation example.

Input Record:

Firstname	Surname	Address1	Address2	Address3	DOB	Telephone
John	O'Brien	3 Oak Rd.	Douglas	Co. Cork	20/4/66	021-234678

Output Record

FN_stan	FN_Soundex	FN_Root	SN_stan	SN_Soundex	SN_root	A1_Num
John	Jon	Jonathon	OBrien	O-165	Brien	3
A1_text	**A1_text_soundex**	**A1_st**	**A2_text**	**A2_str_soundex**	**A3_st**	**A3_text**
Oak	O-200	Road	Douglas	Duglass	County	Cork
DOB_Eur	**DOB_US**	**Telephone**	**Tel_local**			
20041966	04201966	353212346 78	234678			

Grouping Module 9

The aim of the data record grouping process is to significantly speed up the matching step by reducing the number of record pairs which go through the set of complex match scoring routines. This is done by grouping records which have certain similar features – only records within the same group are then compared in the matching phase. (This

greatly reduces the number of matching steps required from NxN to GxHxH where G is the number of groups and H is the number of elements per group).

The module 9 ensures that all actual matches of any record are contained within the same group. The grouping process must be kept simple so that minimal processing time is required to identify elements in the same group. In addition, to have a real impact on efficiency the groups must be substantially smaller than the full dataset (at least 10 times)

After the transformation process is performed on an individual data record a further set of predefined routines is applied to certain fields of the record. These routines extract features from the data fields. These features are included in a small number (2-4) of extra data fields appended to the output record. These labels allow the record to be grouped with other similar records.

The key attributes of the labels are:

- Must be very high probability (99.999%) that all matching records have some or all of the same labels.
- Labels must be easily extracted from the data fields.
- Labels must be impervious to any range data errors which have not been corrected by the transformation process, for example, spelling errors, typing errors, different naming conventions, and mixed fields.

The grouping process is a high speed filtering process to significantly reduce the amount of matches required rather than as a substitute for the matching process. As such, in order to keep the grouping process simple but ensure that no matches are missed, each group is large and the vast majority of records within a group will not match.

An example of the type of routine used in the grouping process is a keyletter routine. The keyletter is defined as the most important matching letter in the field - generally the first letter of the main token – J for John, B for oBrien, O for Oak, D for Douglas, C for